UML 2.0 STATE DIAGRAMS

Concepts & Notations

UML State Diagram

- Used to show the lifetime behavior of a single object
  - What states can it be in?
  - What causes it to move from one state to another?

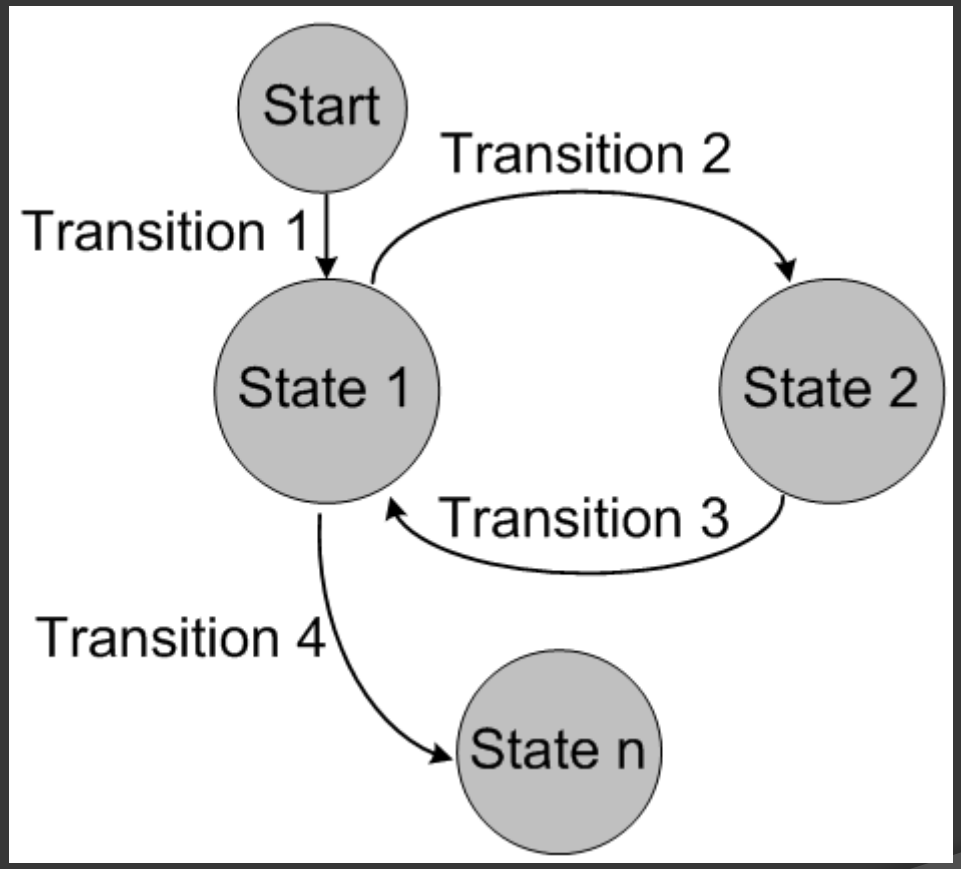
UML State Diagram

- Finite-state Models
- State Machine View
- Events, Transitions, and States
- Examples

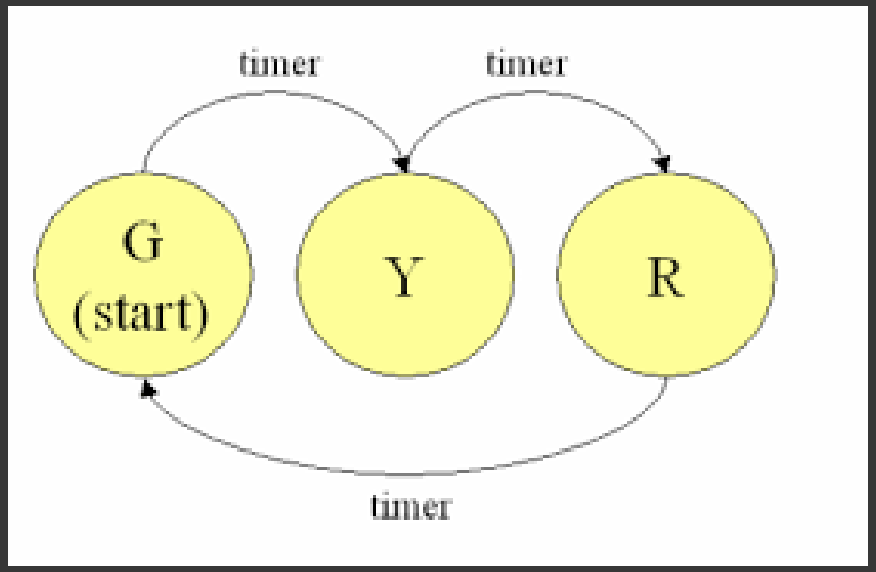
Finite-state Models

- Represent the behavior of a system over time
- Specify control
  - In terms of states, events, and transitions
  - Transitions occur only when they are *enabled*
- Many variants exist
  - Graphical: State charts, UML state diagrams
  - Textual: FSP

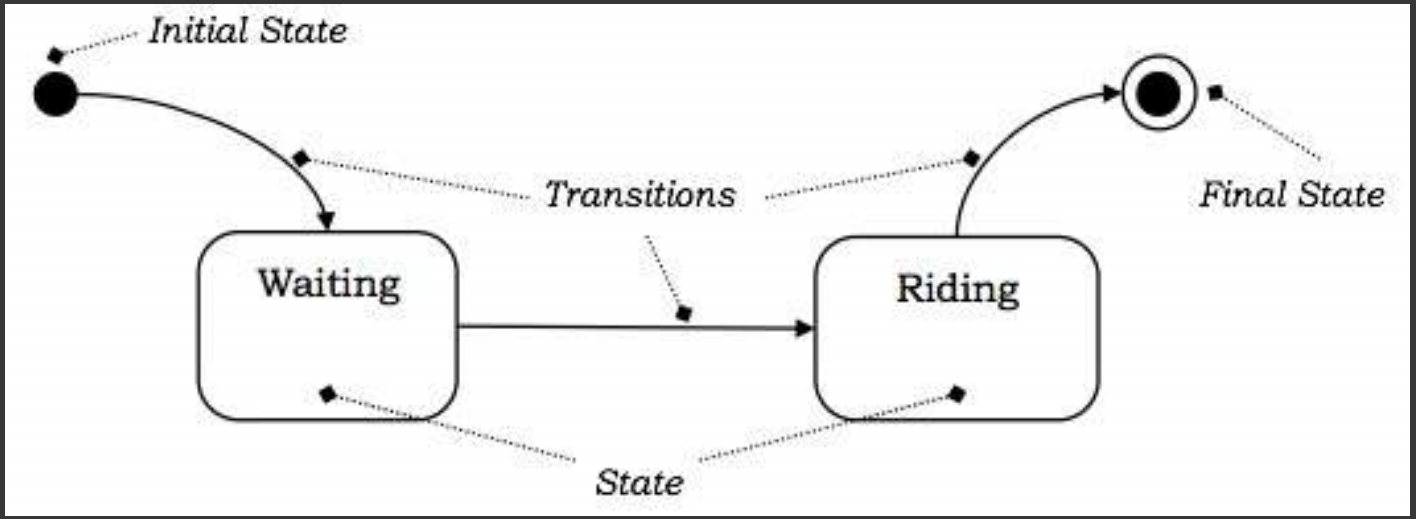
State transition diagrams



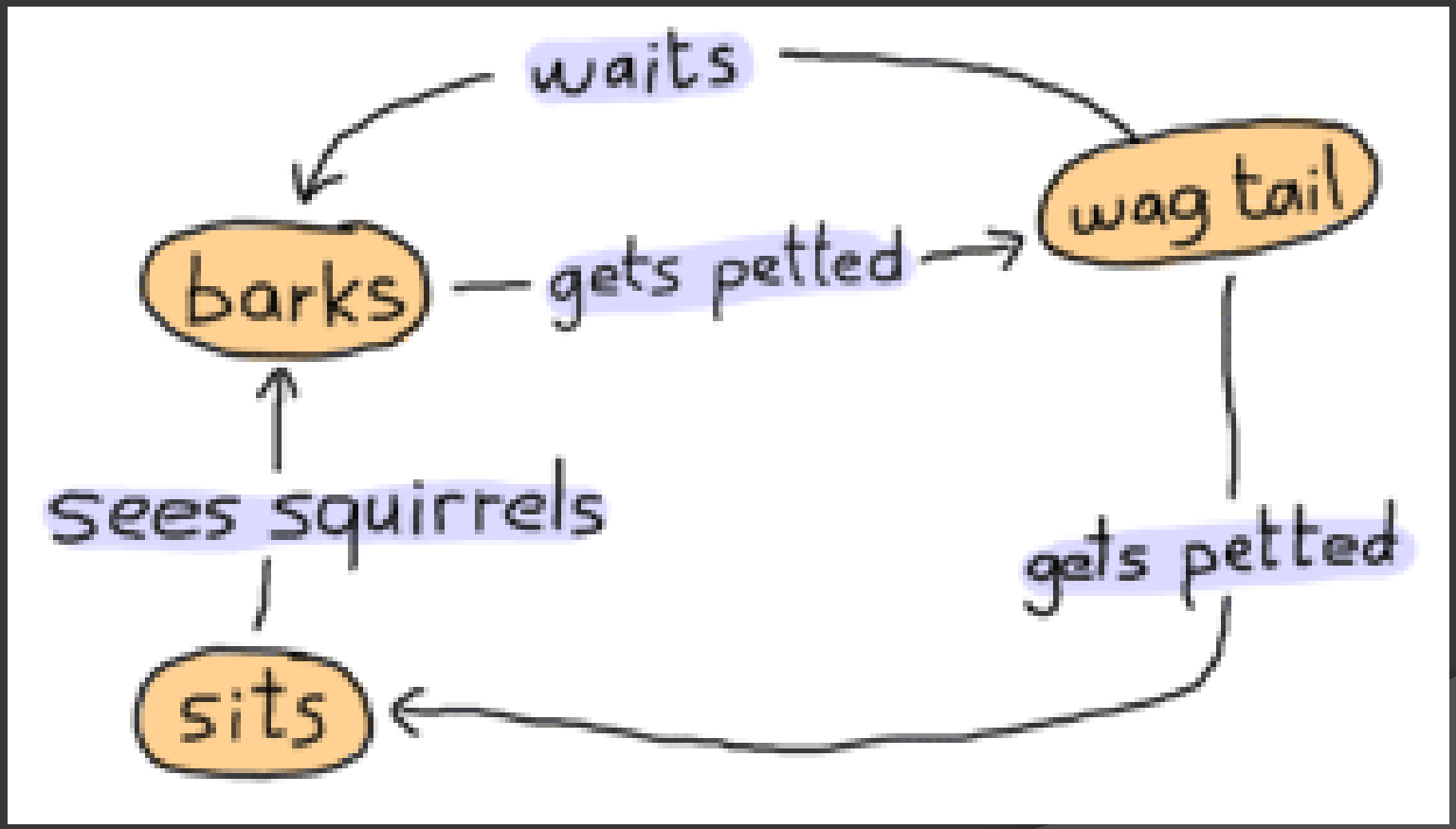
State transition diagrams



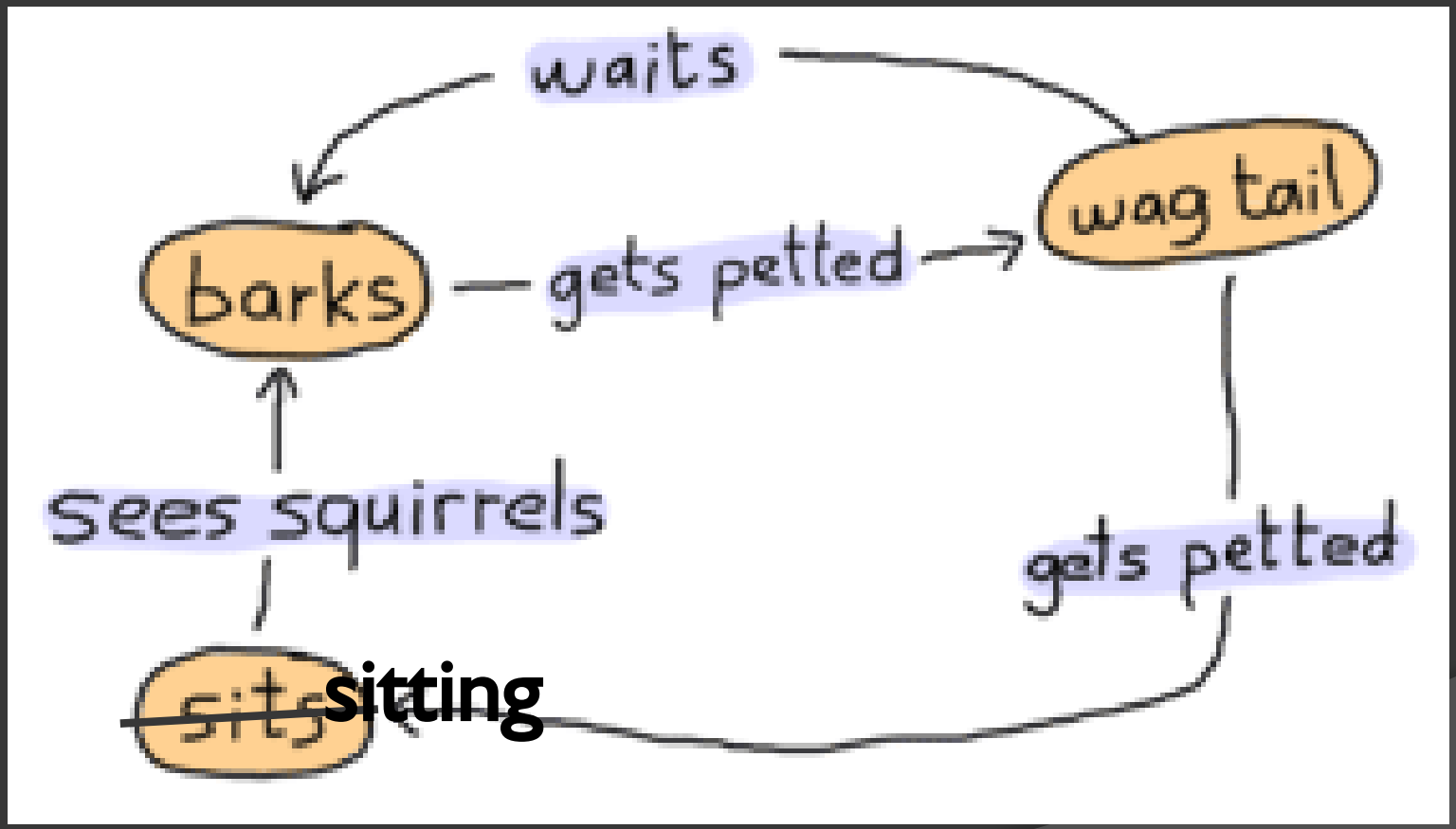
State transition diagrams



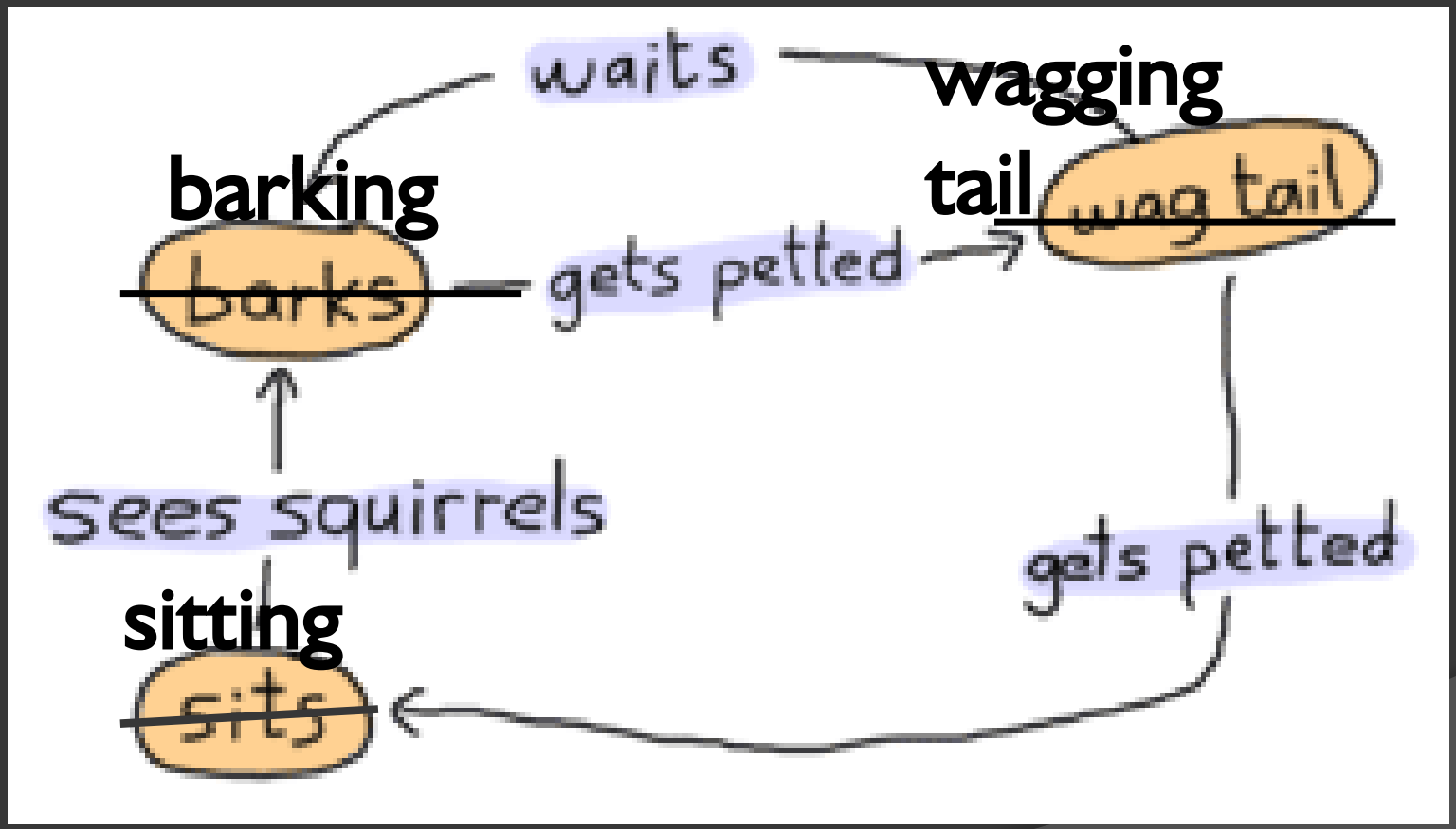
State transition diagrams



State transition diagrams



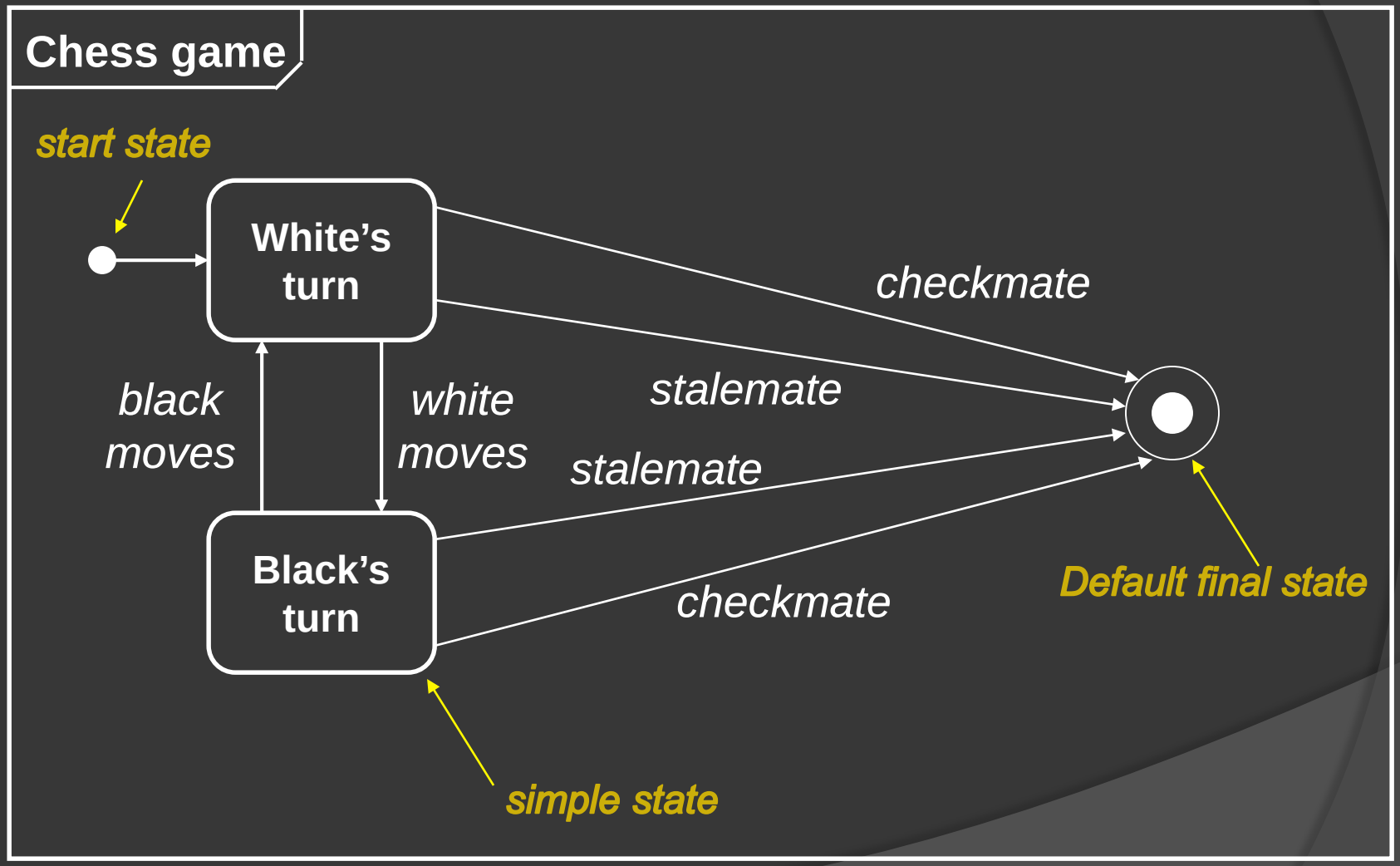
State transition diagrams



State Machine View

- Describes dynamic behavior of objects over time
- Describes states that objects may hold, and how they will react to events when in those states
- Describes classes and corresponding instances
  - Also behaviors, use cases, collaborations
- A localized view
  - External influences summarized as events

Example


Chess game
start state
White's turn
checkmate
black moves
white moves
stalemate
stalemate
Black's turn
checkmate
Default final state
simple state

Event, Transition & State

- Event
  - occurrence at a point in time
  - Instantaneous
  - verbs of past tense
  - onset of a condition
- Transition
- State

Kinds of Events

Type	Description	Syntax
call event	Receipt of an explicit synchronous call request by an object	op (a:T)
change event	A change in value of a Boolean expression	when (exp)
signal event	Receipt of an explicit, named, asynchronous communication among objects	sname (a:T)
time event	The arrival of an absolute time or the passage of a relative amount of time	after (time)

Kinds of Events

Type	Description	Syntax
call event	**Receipt of an explicit synchronous call request by an object**	**op (a:T)**
change event	A change in value of a Boolean expression	when (exp)
signal event	Receipt of an explicit, named, asynchronous communication among objects	sname (a:T)
time event	The arrival of an absolute time or the passage of a relative amount of time	after (time)

Kinds of Events

Type	Description	Syntax
call event	Receipt of an explicit synchronous call request by an object	op (a:T)
change event	**A change in value of a Boolean expression**	**when (exp)**
signal event	Receipt of an explicit, named, asynchronous communication among objects	sname (a:T)
time event	The arrival of an absolute time or the passage of a relative amount of time	after (time)

Kinds of Events

Type	Description	Syntax
call event	Receipt of an explicit synchronous call request by an object	op (a:T)
change event	A change in value of a Boolean expression	when (exp)
signal event	**Receipt of an explicit, named, asynchronous communication among objects**	**sname (a:T)**
time event	The arrival of an absolute time or the passage of a relative amount of time	after (time)

Kinds of Events

Type	Description	Syntax
call event	Receipt of an explicit synchronous call request by an object	op (a:T)
change event	A change in value of a Boolean expression	when (exp)
signal event	Receipt of an explicit, named, asynchronous communication among objects	sname (a:T)
time event	**The arrival of an absolute time or the passage of a relative amount of time**	**after (time)**

Event, Transition & State

- Event
- Transition
  - Instantaneous change in state
  - 4 key elements
    - event trigger
    - guard condition
    - effect
    - target state
- State

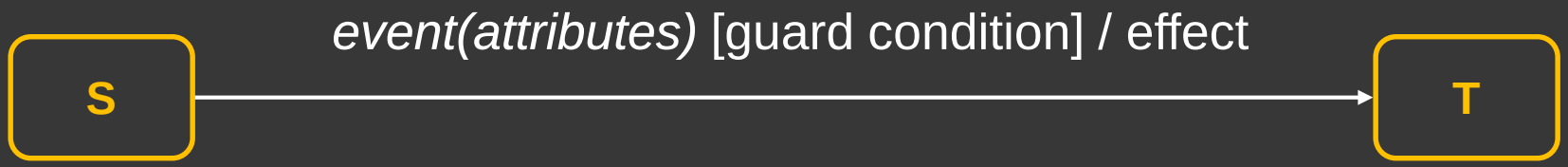
Kinds of Transitions

Transition Type	Key point	Syntax
entry transition	do setup	entry / effect
exit transition	do clean up	exit / effect
external transition	include entry / exit activity	e(a:T) [guard condition] / effect
internal transition	exclude entry / exit activity	e(a:T) [guard condition] / effect

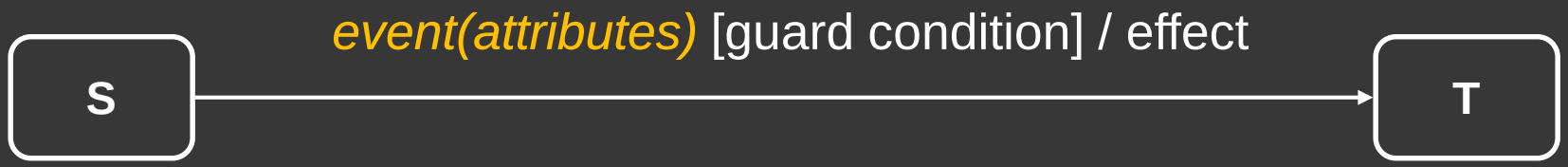
Kinds of Transitions

Transition Type	Key point	Syntax
entry transition	do setup	entry / effect
exit transition	do clean up	exit / effect
external transition	**include entry / exit activity**	**e(a:T) [guard condition] / effect**
internal transition	exclude entry / exit activity	e(a:T) [guard condition] / effect

Transition

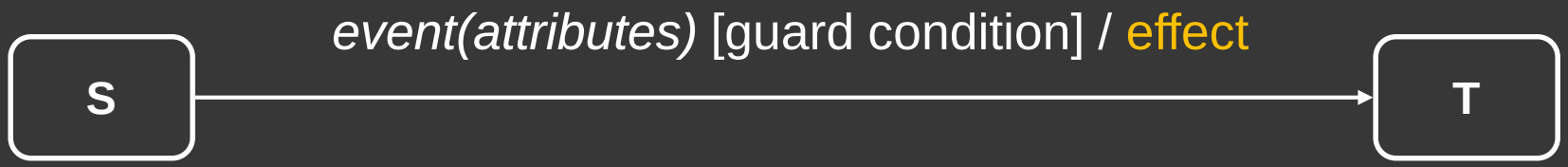


Transition



Transition

Transition



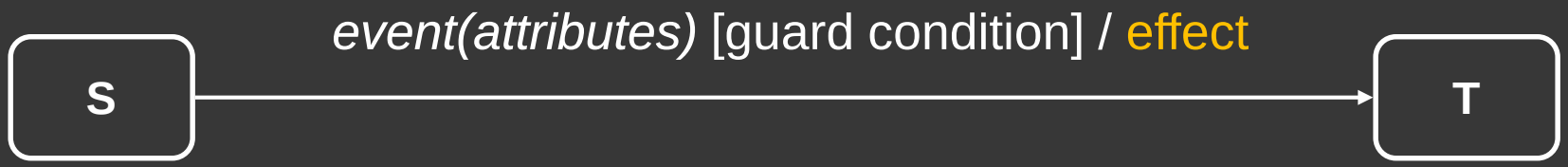
Effects

- An effect may be a (regular) **activity** or a do-activity.
  - activity: a primitive computation
    - x = 3
    - z = x+1
    - sending a signal
    - calling an operation
    - new Widget(x,3)
    - widg.getValue()
    - widg.setValue(5);

Effects

- An effect may be an **activity** or a do-activity.
  - do-activity : a list of simpler actions or activities
    - the specification of executable behavior as the coordinated sequential and concurrent execution of subordinate units.

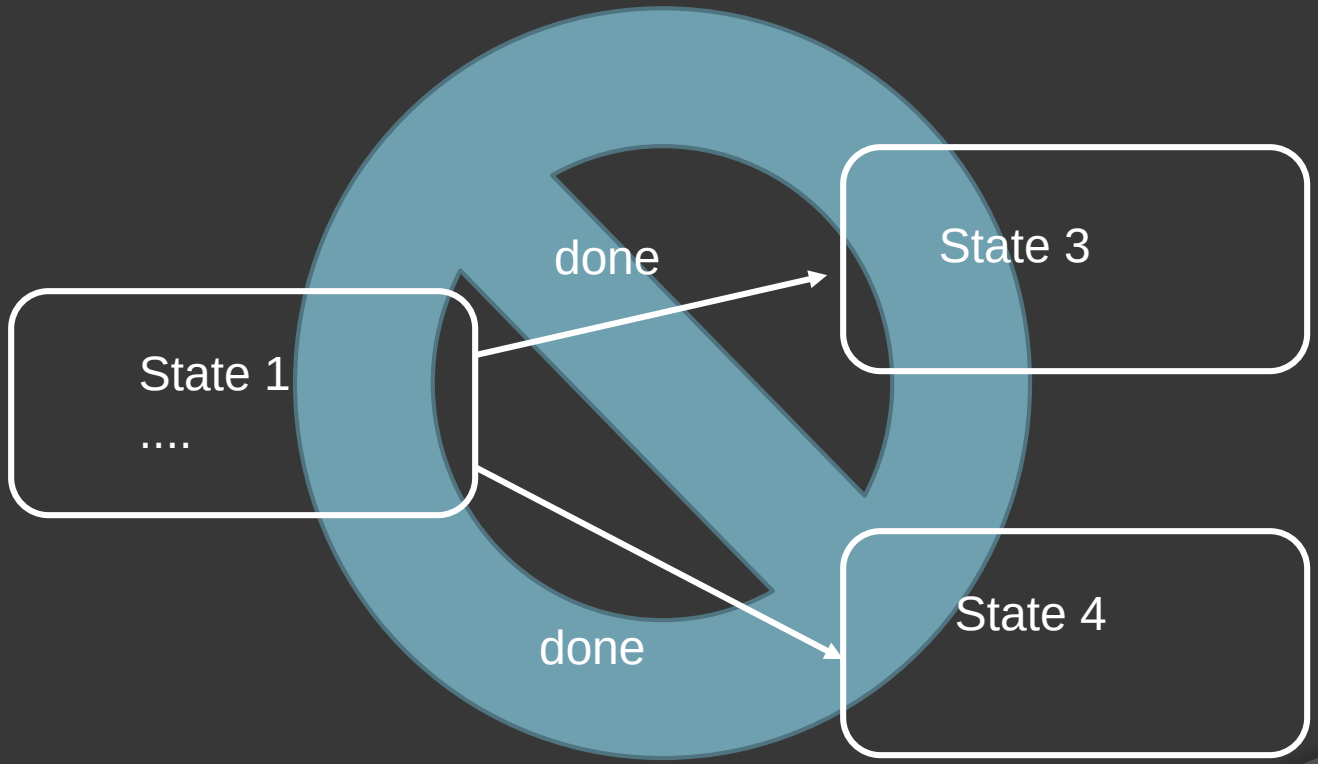
Transition



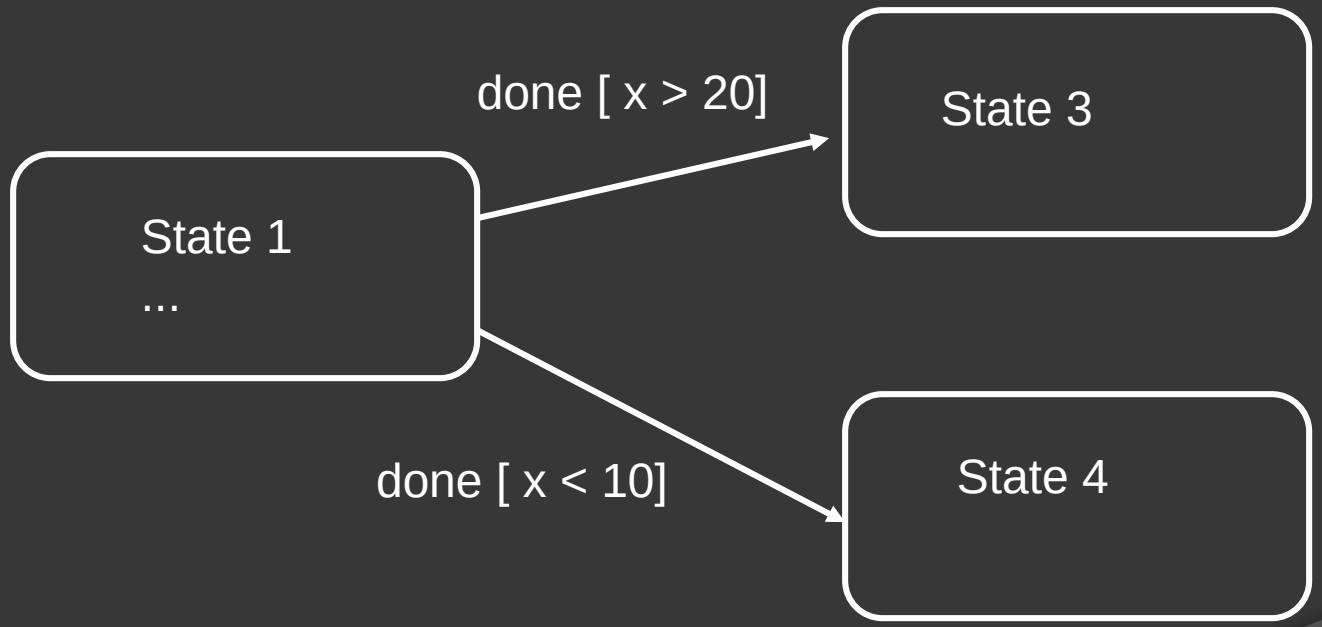
Multiple transitions

- Multiple transitions from the same state may specify the same event trigger
  - only one transition may fire in response to one event occurrence
  - each transition with same event must have a different guard condition
  - Often, conditions together cover all possibilities
  - If an uncovered possibility occurs, the event is ignored

Not allowed ...



Uncovered possibilities ...



Completion transition

- A transition that lacks an explicit trigger event
  - is triggered by the completion of activity in the state that it leaves
  - may have a guard condition
  - takes priority over ordinary events

Kinds of Transitions

Transition Type	Key point	Syntax
entry transition	**do setup**	**entry / effect**
exit transition	do clean up	exit / effect
external transition	include entry / exit activity	e(a:T) [guard condition] / effect
internal transition	exclude entry / exit activity	e(a:T) [guard condition] / effect

Kinds of Transitions

Transition Type	Key point	Syntax
entry transition	do setup	entry / effect
exit transition	**do clean up**	**exit / effect**
external transition	include entry / exit activity	e(a:T) [guard condition] / effect
internal transition	exclude entry / exit activity	e(a:T) [guard condition] / effect

Kinds of Transitions

Transition Type	Key point	Syntax
entry transition	do setup	entry / effect
exit transition	do clean up	exit / effect
external transition	include entry / exit activity	e(a:T) [guard condition] / effect
internal transition	**exclude entry / exit activity**	**e(a:T) [guard condition] / effect**

Examples

figure 1

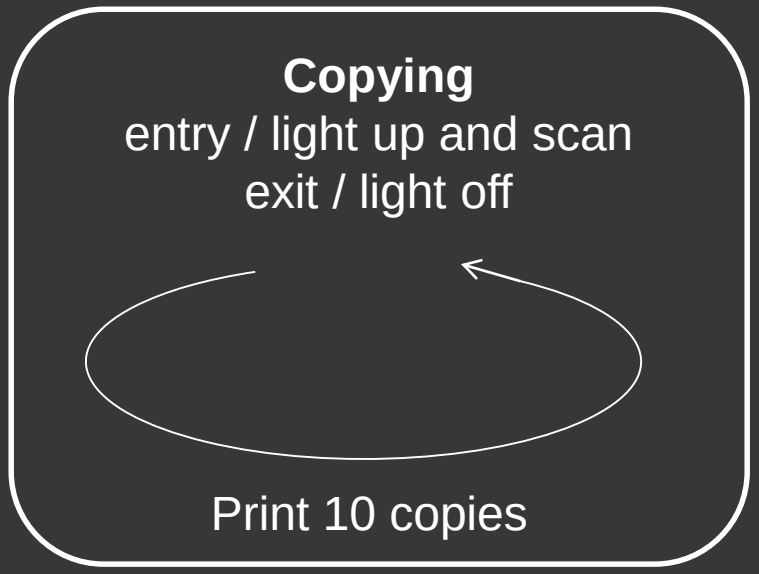


figure 2

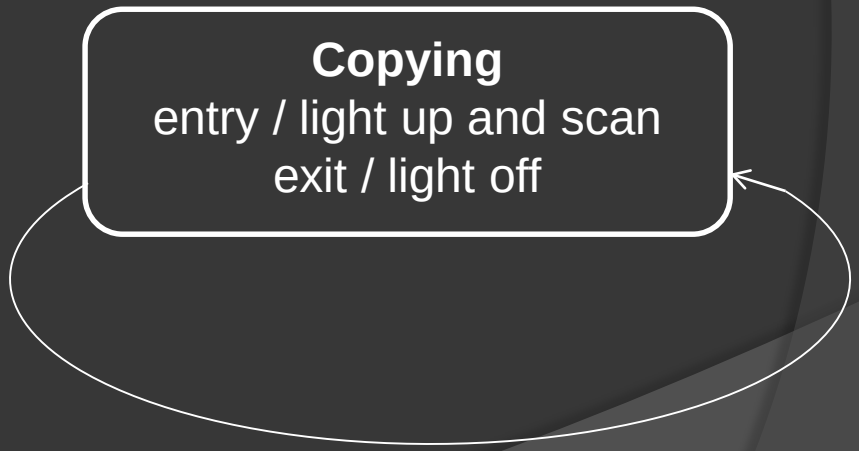


figure 3

Order of activities

1. activities on incoming transition
2. entry activities
3. do-activities
   - execution of behavior within a state machine that is based on holding a given state
   - starts when the state is entered
   - continues until the activity completes on its own or the state is exited
   - not terminated by the firing of an internal transition
4. exit activities
5. activities on outgoing transition

Event, State & Transition

- Event
- Transition
- State
  - behavioral condition that persists in time
  - corresponds to verbs with suffix of “-ing”
  - an abstraction of values of attributes and configuration of objects
    - a set of object with similar values
    - object waits for events
    - object performs ongoing activity

Kinds of States

- simple state
- notation:

Kinds of States

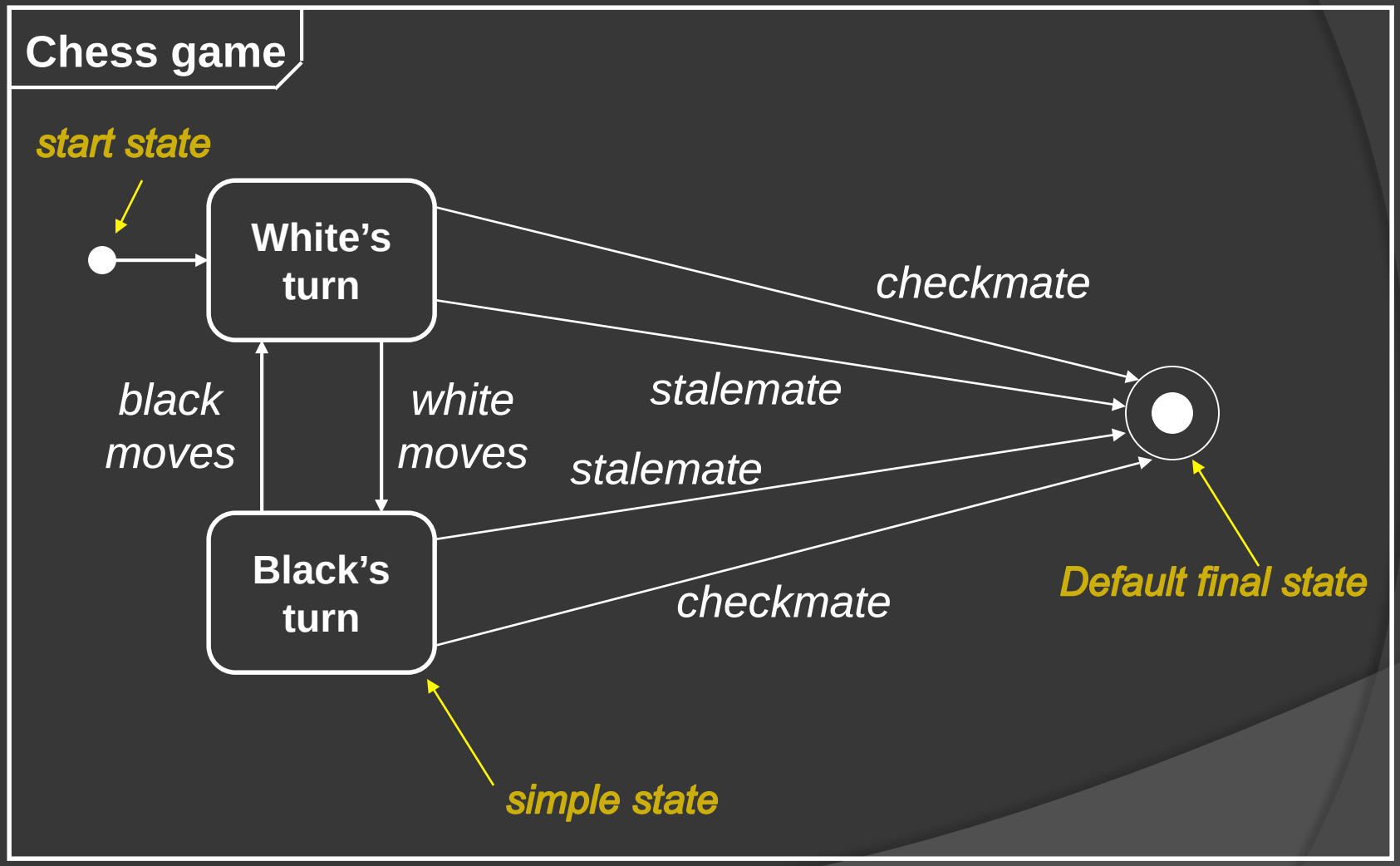
- initial state

- notation:

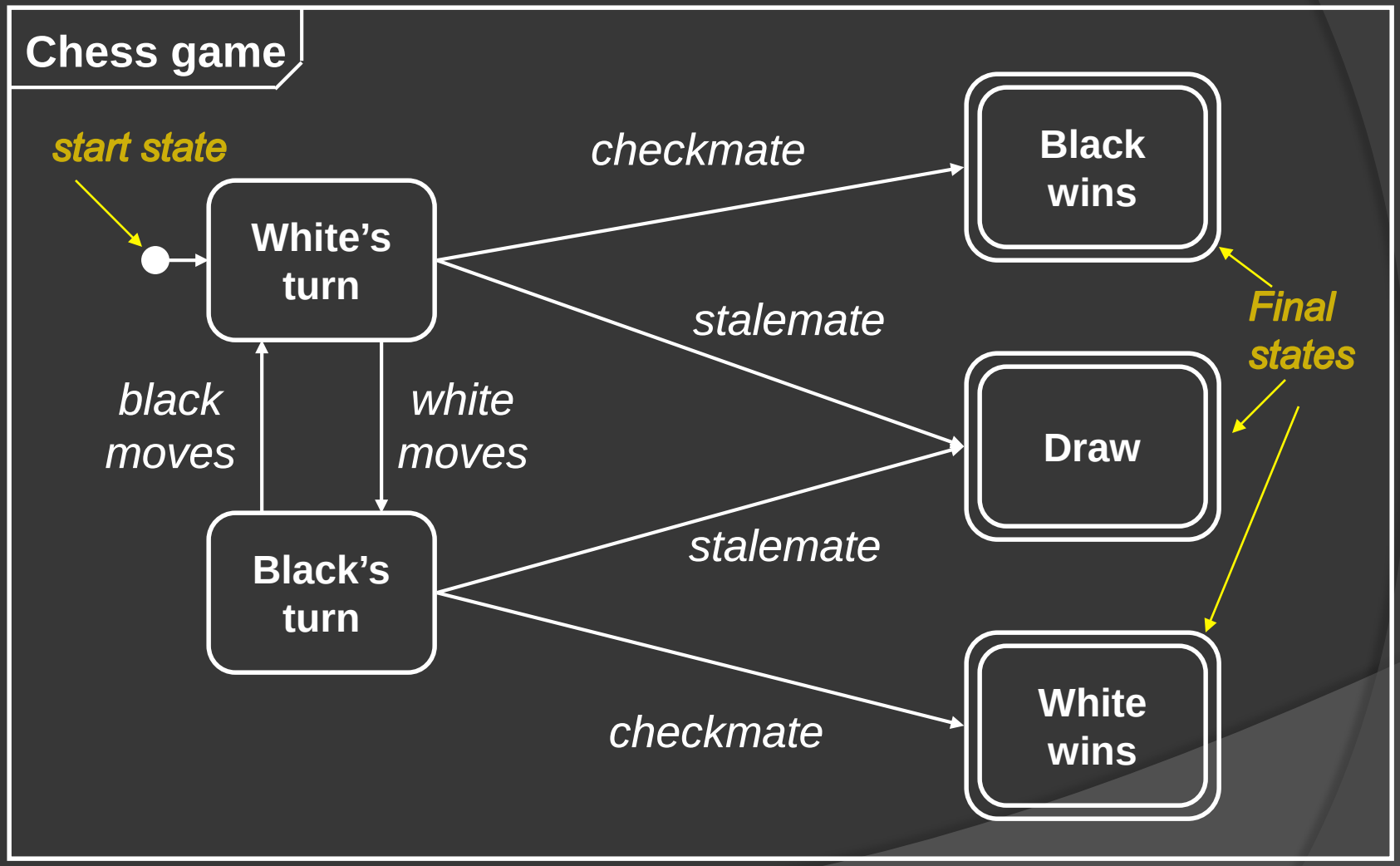
Kinds of States

- final state
- notation:

Example

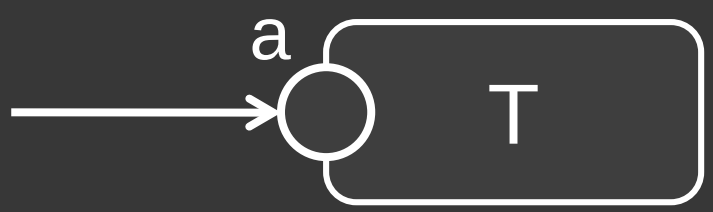

Chess game
start state
White's turn
checkmate
black moves
white moves
stalemate
stalemate
Black's turn
checkmate
Default final state
simple state

Example


Chess game
start state
White's turn
checkmate
Black wins
stalemate
Final states
black moves
white moves
Draw
Black's turn
stalemate
checkmate
White wins

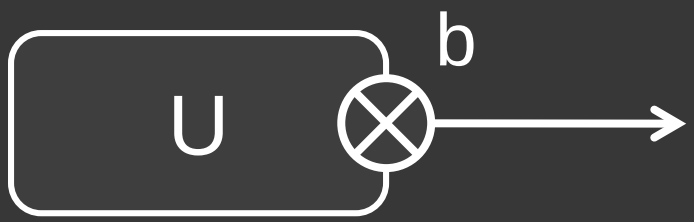
Kinds of States

- entry point
- notation:

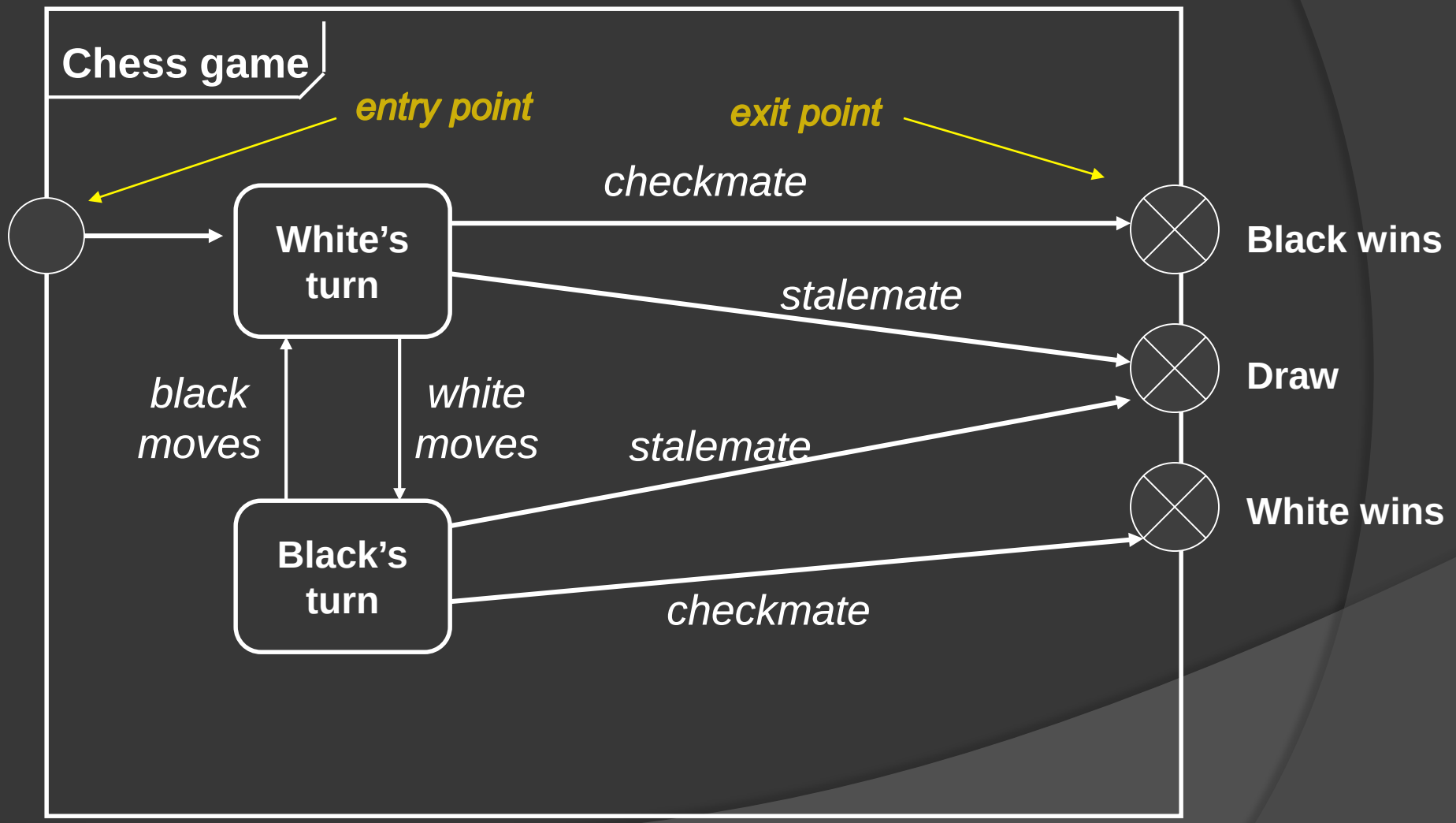


Kinds of States

- exit point
- notation:



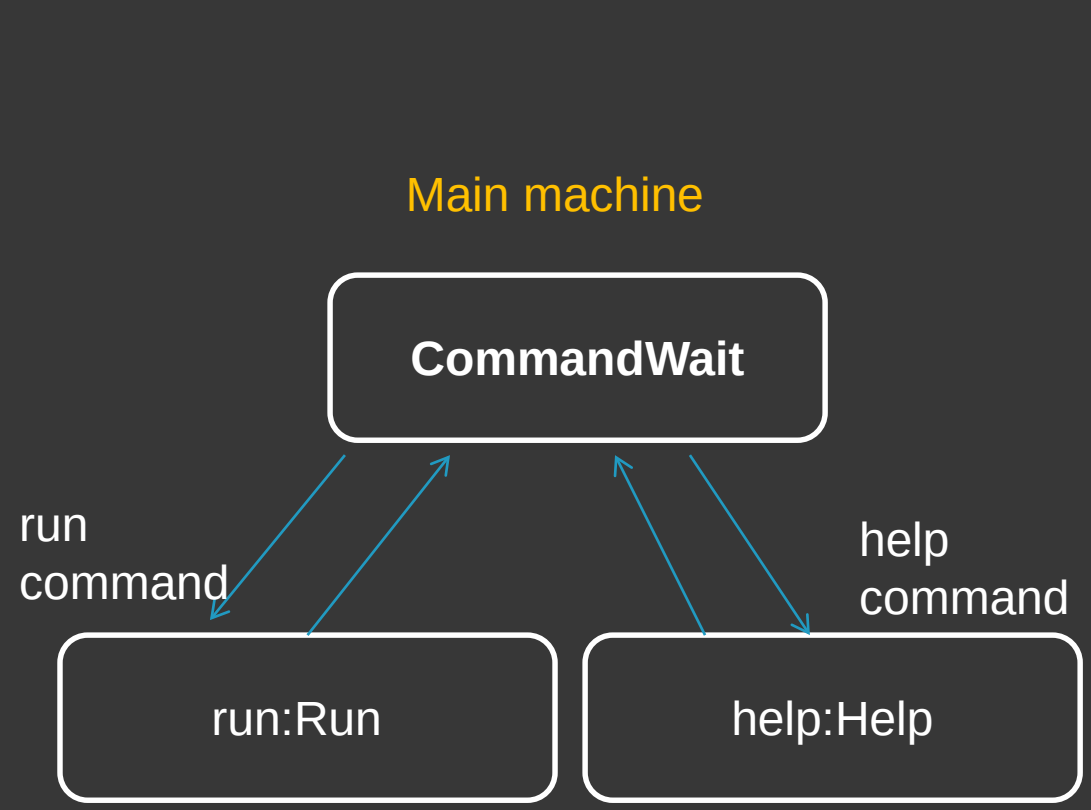
Example

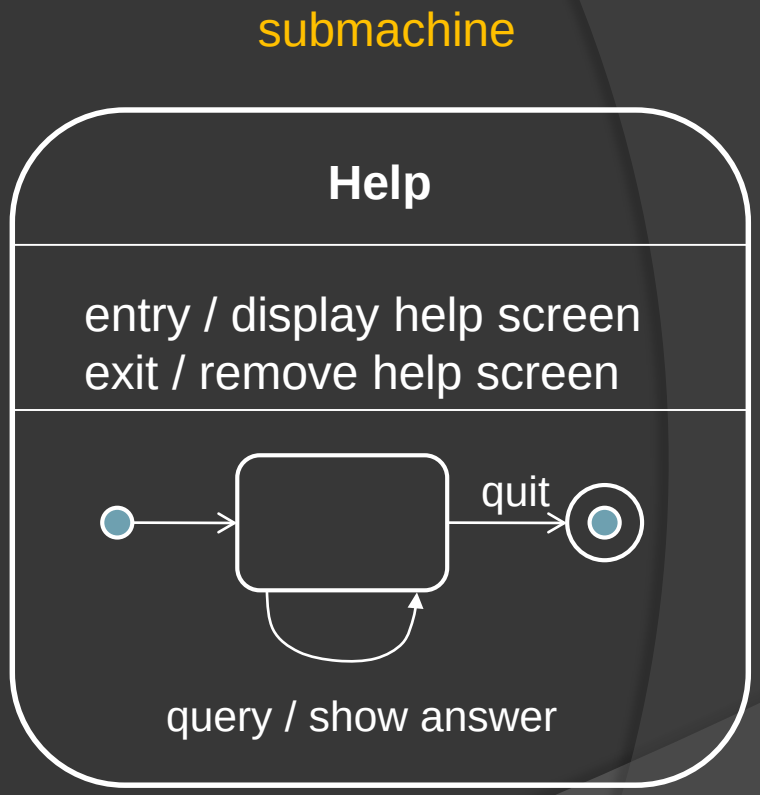

Chess game
entry point
exit point
checkmate
White's turn
Black wins
stalemate
Draw
black moves
white moves
stalemate
White wins
Black's turn
checkmate

Kinds of States

- submachine state
- notation:

Examples





submachine state

submachine state

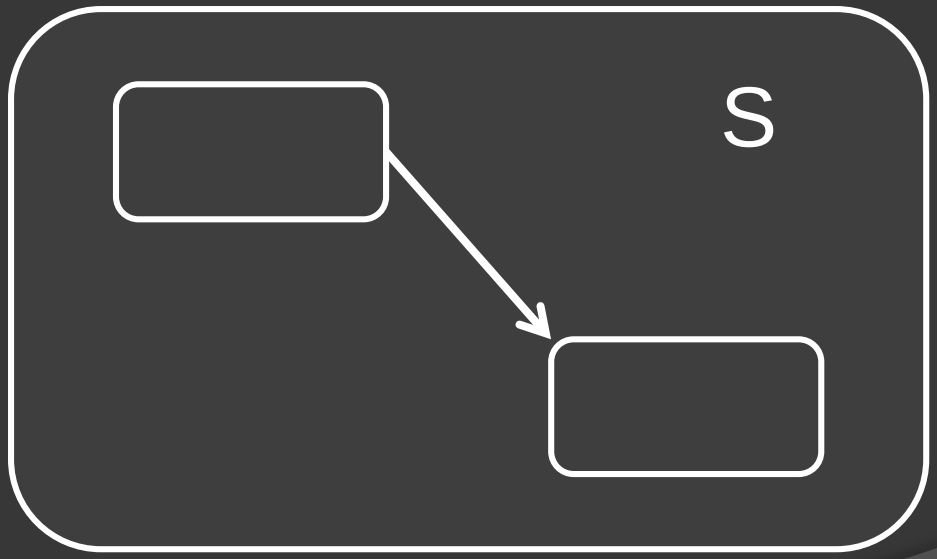
This submachine can be used many times

Composite State

- Introduces an abstract “Superstate”
  - decomposed into multiple sub-states
  - when the superstate is active, exactly one of its sub-states is active

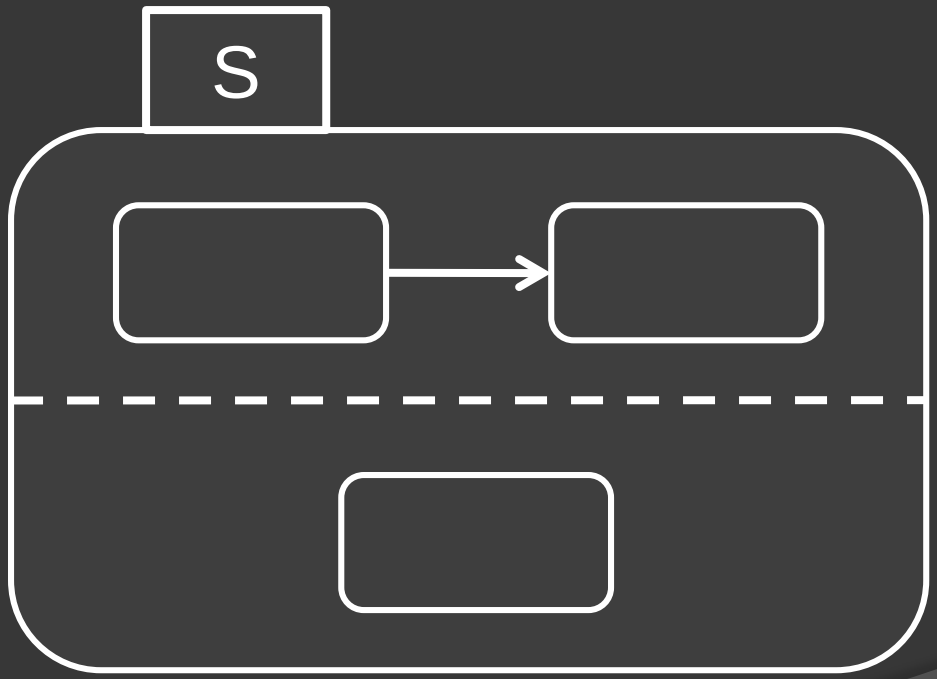
Composite State

- nonorthogonal state

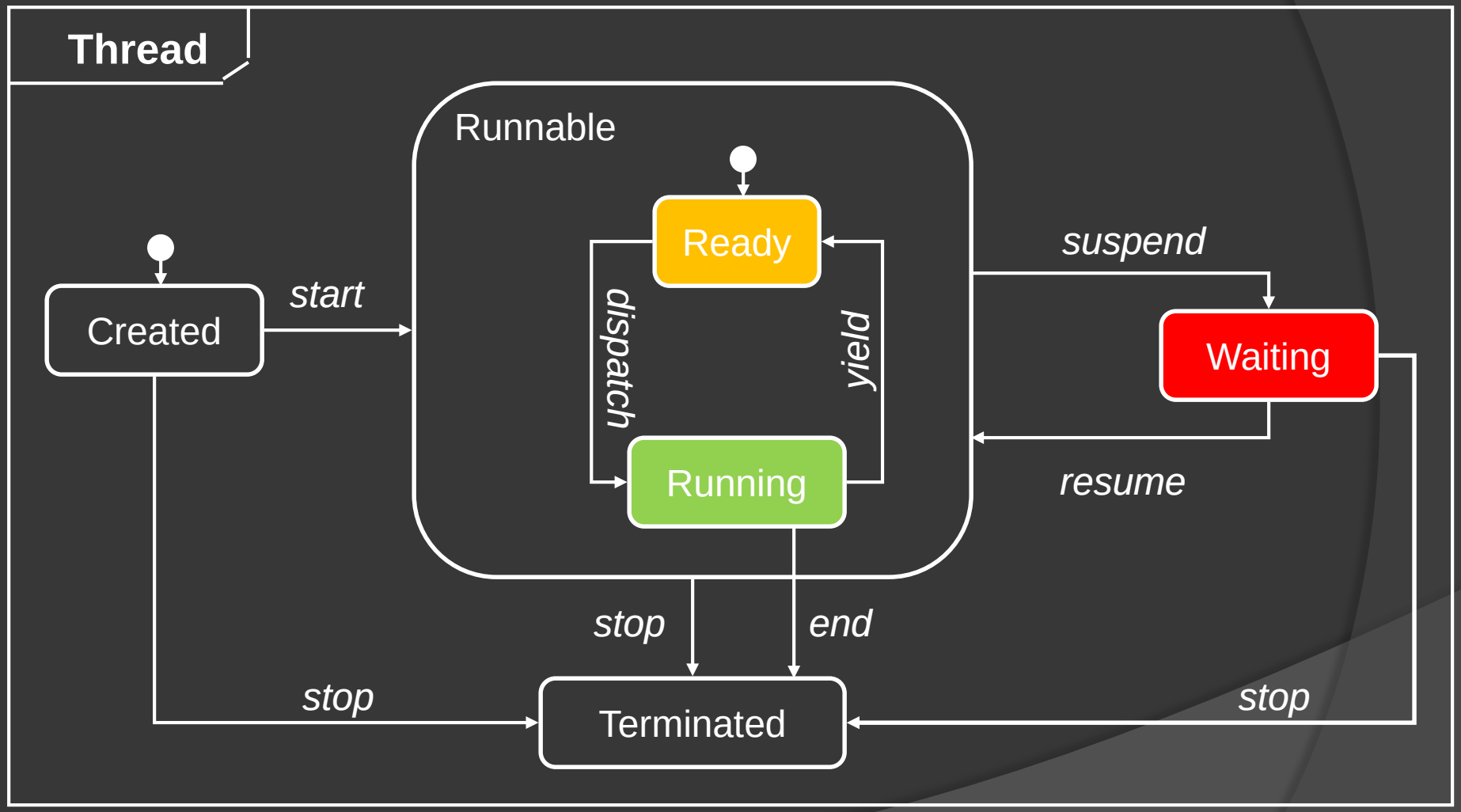


Composite State

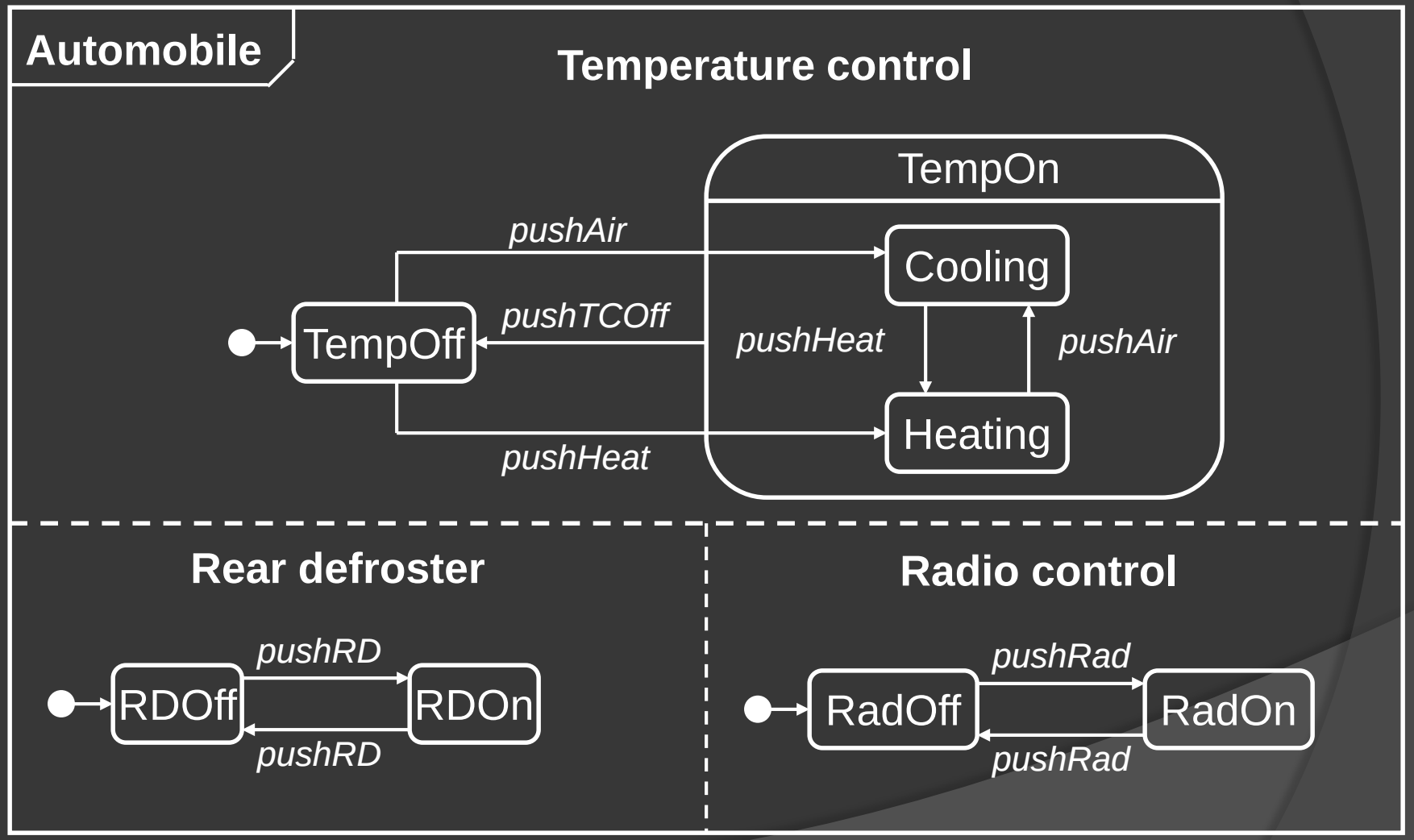
- orthogonal state



Examples


Thread
Runnable
Ready
Running
dispatch
yield
Created
start
suspend
Waiting
resume
stop
end
stop
Terminated
stop

Examples


Automobile
Temperature control
TempOn
pushAir
Cooling
TempOff
pushTCOff
pushHeat
pushAir
Heating
pushHeat
Rear defroster
pushRD
RDOff
RDOn
pushRD
Radio control
pushRad
RadOff
RadOn
pushRad

Review

- States, Events, Transitions
  - Firing rules
- Effects: actions and activities
  - Ordering of activities
- Simple, submachine, and composite states
  - orthogonal
  - non-orthogonal